Assignment 11.2 Assignment on Machine Learning

Basak Atanu

05-27-2022

BINARY CLASSIFIER DATA.

```
setwd("C:\\Users\\atanu\\Documents\\BellevueUniversity_MSDS\\DSC520\\Repository\\dsc520_")
binary_data <- read.csv("data\\binary-classifier-data.csv")
head(binary_data)
```

```
##   label        x        y
## 1     0 70.88469 83.17702
## 2     0 74.97176 87.92922
## 3     0 73.78333 92.20325
## 4     0 66.40747 81.10617
## 5     0 69.07399 84.53739
## 6     0 72.23616 86.38403
```

Let split the data for training and test to see how fitted model work in test.

```
library(caTools)
library(class)
split <- sample.split(binary_data, SplitRatio = 0.7)
train_binary_data <- subset(binary_data, split == "TRUE")
test_binary_data <- subset(binary_data, split == "FALSE")
```

Lets plot the scatter diagram of the data.

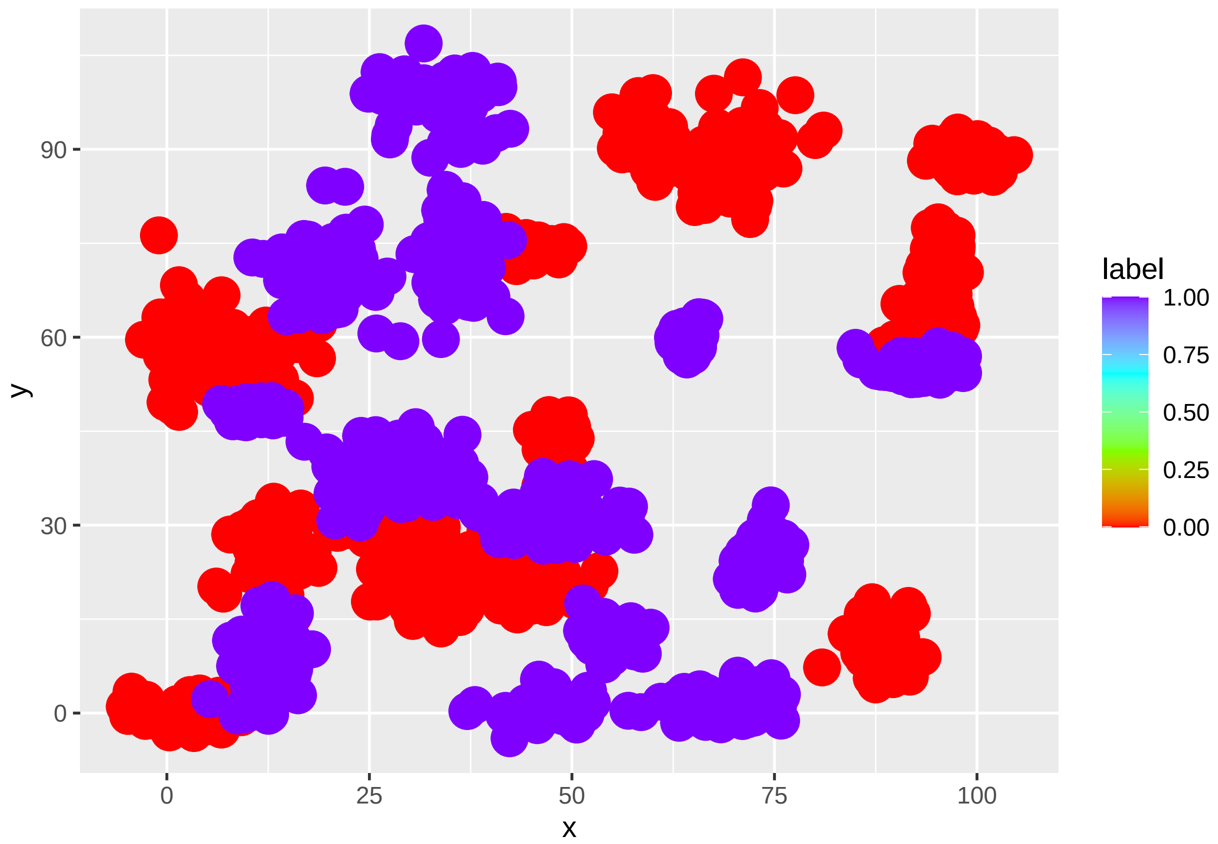
```
library(ggplot2)
library(hrbrthemes)
```

```
## NOTE: Either Arial Narrow or Roboto Condensed fonts are required to use these themes.
```

```
##       Please use hrbrthemes::import_roboto_condensed() to install Roboto Condensed and
```

```
##       if Arial Narrow is not on your system, please see https://bit.ly/arialnarrow
```

```
ggplot(binary_data, aes(x=x, y=y, color=label)) + geom_point(size=6) + scale_colour_gradientn(colours=ra
```



Lets fit the KNN for different K value and callculate the corresponding accuracy. ## K=3

```
classifier_knn_3 <- knn(train = train_binary_data,test = test_binary_data,cl = train_binary_data$label,k

cm <- table(test_binary_data$label, classifier_knn_3)
cm
```

```
##    classifier_knn_3
##       0   1
##   0 248   7
##   1   5 239
```

```
misClassError <- mean(classifier_knn_3 != test_binary_data$label)
accuracy_3 = 1-misClassError
print(paste('Accuracy =', accuracy_3))
```

```
## [1] "Accuracy = 0.975951903807615"
```

K=5

```
classifier_knn_5 <- knn(train = train_binary_data,test = test_binary_data,cl = train_binary_data$label,l

cm <- table(test_binary_data$label, classifier_knn_5)
cm
```

```
##    classifier_knn_5
##       0   1
##   0 249   6
##   1   5 239
```

```
misClassError <- mean(classifier_knn_5 != test_binary_data$label)
accuracy_5 = 1-misClassError
print(paste('Accuracy =', accuracy_5))
```

```
## [1] "Accuracy = 0.977955911823647"
```

K=10

```
classifier_knn_10 <- knn(train = train_binary_data,test = test_binary_data,cl = train_binary_data$label

cm <- table(test_binary_data$label, classifier_knn_10)
cm
```

```
##    classifier_knn_10
##       0   1
##   0 248   7
##   1   5 239
```

```
misClassError <- mean(classifier_knn_10 != test_binary_data$label)
accuracy_10 = 1-misClassError
print(paste('Accuracy =', accuracy_10))
```

```
## [1] "Accuracy = 0.975951903807615"
```

K=15

```
classifier_knn_15 <- knn(train = train_binary_data,test = test_binary_data,cl = train_binary_data$label

cm <- table(test_binary_data$label, classifier_knn_15)
cm
```

```
##    classifier_knn_15
##       0   1
##   0 247   8
##   1   6 238
```

```r
misClassError <- mean(classifier_knn_15 != test_binary_data$label)
accuracy_15 = 1-misClassError
print(paste('Accuracy =', accuracy_15))
```

```
## [1] "Accuracy = 0.971943887775551"
```

K=20

```r
classifier_knn_20 <- knn(train = train_binary_data,test = test_binary_data,cl = train_binary_data$label


cm <- table(test_binary_data$label, classifier_knn_20)
cm
```

```
##    classifier_knn_20
##       0   1
##   0 248   7
##   1   5 239
```

```r
misClassError <- mean(classifier_knn_20 != test_binary_data$label)
accuracy_20 = 1-misClassError
print(paste('Accuracy =', accuracy_20))
```

```
## [1] "Accuracy = 0.975951903807615"
```

K=25

```r
classifier_knn_25 <- knn(train = train_binary_data,test = test_binary_data,cl = train_binary_data$label

cm <- table(test_binary_data$label, classifier_knn_25)
cm
```

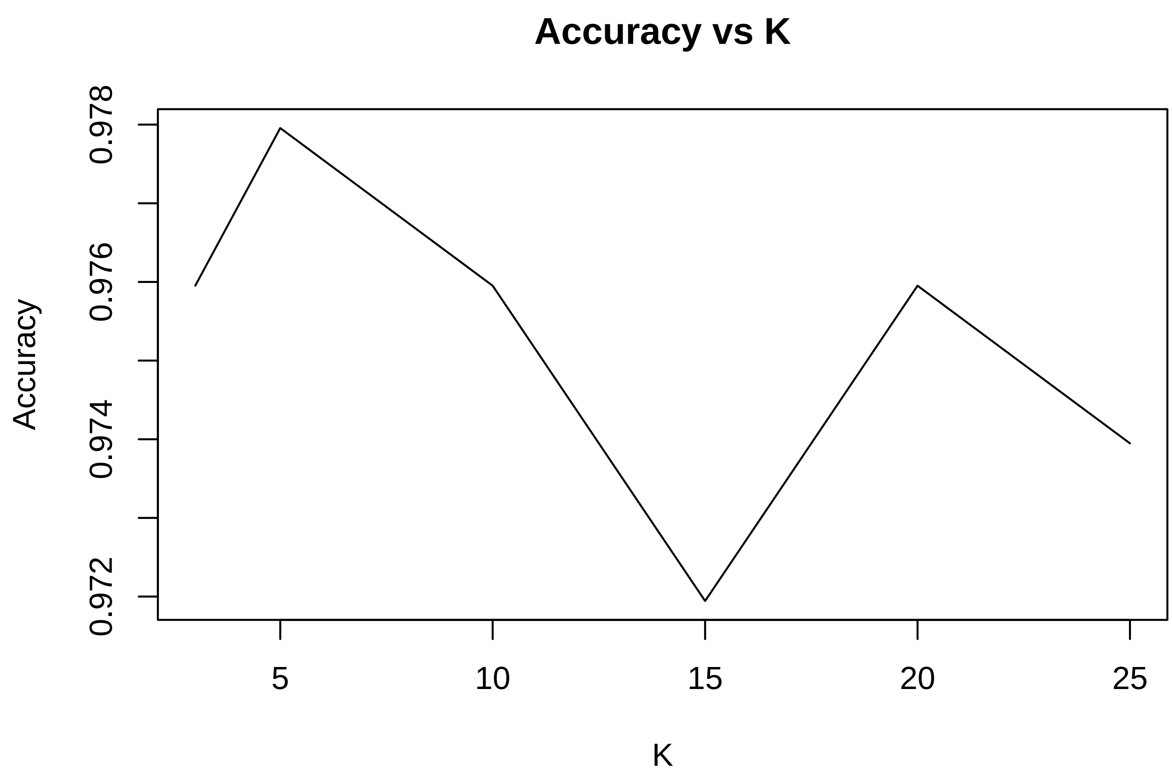
```
##    classifier_knn_25
##       0   1
##   0 247   8
##   1   5 239
```

```r
misClassError <- mean(classifier_knn_25 != test_binary_data$label)
accuracy_25 = 1-misClassError
print(paste('Accuracy =', accuracy_25))
```

```
## [1] "Accuracy = 0.973947895791583"
```

Plot for accuray for different K value.

```
k <- c(3,5,10,15,20,25)
accuracy <- c(accuracy_3,accuracy_5,accuracy_10, accuracy_15, accuracy_20, accuracy_25)
accuracy_by_k <- data.frame(k,accuracy)
plot(accuracy_by_k,type="l",ylab="Accuracy",
 xlab="K",main="Accuracy vs K")
```



As per the plot we can say that K= 5 is giving the optimal accuracy for different k values. # ———— ———————————— # TRINARY CLASSIFIER DATA. # ————————————————

```
setwd("C:\\Users\\atanu\\Documents\\BellevueUniversity_MSDS\\DSC520\\Repository\\dsc520_")
trinary_data <- read.csv("data\\trinary-classifier-data.csv")
head(trinary_data)
```

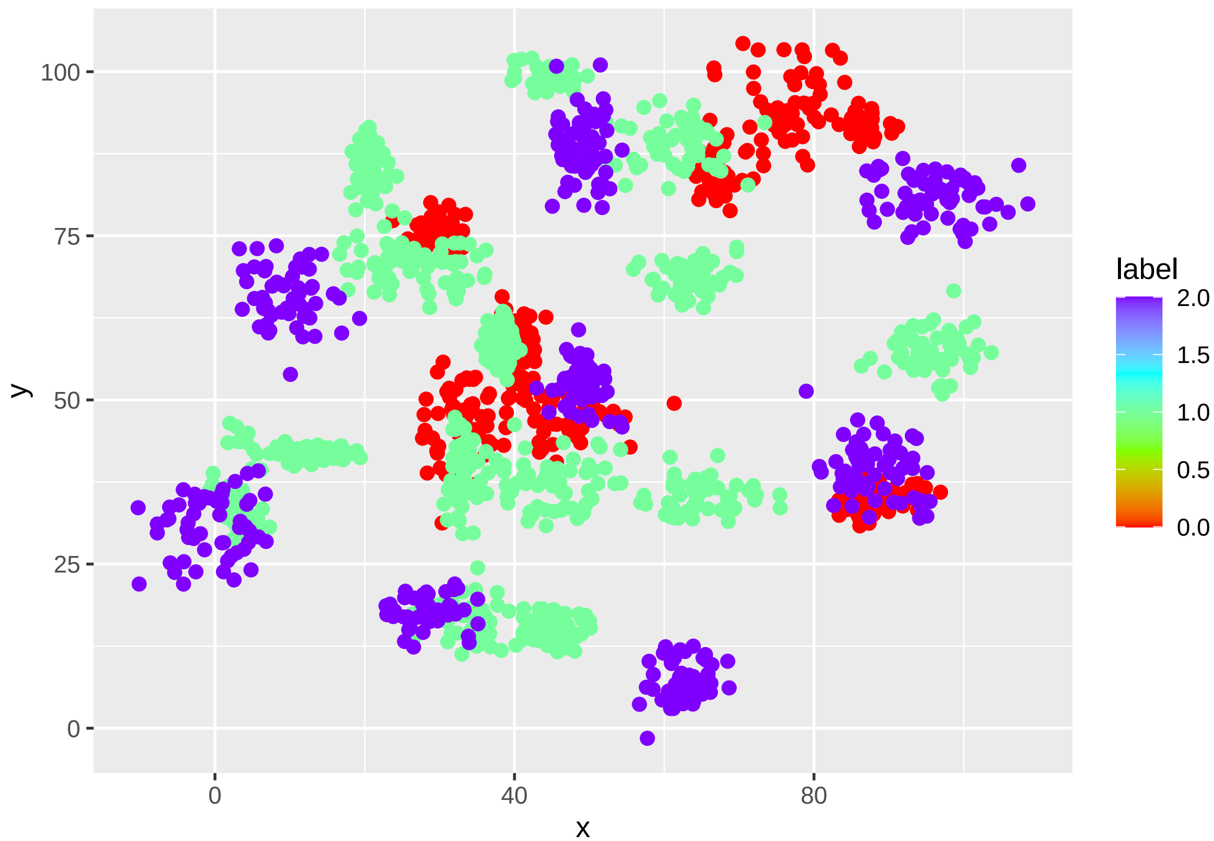
```
##   label        x        y
## 1     0 30.08387 39.63094
## 2     0 31.27613 51.77511
## 3     0 34.12138 49.27575
## 4     0 32.58222 41.23300
## 5     0 34.65069 45.47956
## 6     0 33.80513 44.24656
```

Let split the data for training and test to see how fitted model work in test.

```
library(caTools)
library(class)
split <- sample.split(trinary_data, SplitRatio = 0.7)
train_trinary_data <- subset(trinary_data, split == "TRUE")
test_trinary_data <- subset(trinary_data, split == "FALSE")
```

Lets plot the scatter diagram of the data.

```
library(ggplot2)
library(hrbrthemes)
ggplot(trinary_data, aes(x=x, y=y, color=label)) + geom_point(size=2) + scale_colour_gradientn(colours=r
```



Lets fit the KNN for different K value and callculate the corresponding accuracy. ## K=3

```
classifier_knn_3 <- knn(train = train_trinary_data,test = test_trinary_data,cl = train_trinary_data$labe

cm <- table(test_trinary_data$label, classifier_knn_3)
cm
```

```
##    classifier_knn_3
##       0   1   2
##   0 123   9   0
##   1   4 225  11
##   2   2  12 137
```

```r
misClassError <- mean(classifier_knn_3 != test_trinary_data$label)
accuracy_3 = 1-misClassError
print(paste('Accuracy =', accuracy_3))
```

```
## [1] "Accuracy = 0.927342256214149"
```

K=5

```r
classifier_knn_5 <- knn(train = train_trinary_data,test = test_trinary_data,cl = train_trinary_data$labe

cm <- table(test_trinary_data$label, classifier_knn_5)
cm
```

```
##    classifier_knn_5
##       0   1   2
##   0 125   7   0
##   1   3 226  11
##   2   4  13 134
```

```r
misClassError <- mean(classifier_knn_5 != test_trinary_data$label)
accuracy_5 = 1-misClassError
print(paste('Accuracy =', accuracy_5))
```

```
## [1] "Accuracy = 0.927342256214149"
```

K=10

```r
classifier_knn_10 <- knn(train = train_trinary_data,test = test_trinary_data,cl = train_trinary_data$lab

cm <- table(test_trinary_data$label, classifier_knn_10)
cm
```

```
##    classifier_knn_10
##       0   1   2
##   0 121  11   0
##   1  10 220  10
##   2   8  13 130
```

```r
misClassError <- mean(classifier_knn_10 != test_trinary_data$label)
accuracy_10 = 1-misClassError
print(paste('Accuracy =', accuracy_10))
```

```
## [1] "Accuracy = 0.90057361376673"
```

K=15

```r
classifier_knn_15 <- knn(train = train_trinary_data,test = test_trinary_data,cl = train_trinary_data$la

cm <- table(test_trinary_data$label, classifier_knn_15)
cm
```

```
##    classifier_knn_15
##       0   1   2
##   0 112  19   1
##   1  13 216  11
##   2   9  13 129
```

```r
misClassError <- mean(classifier_knn_15 != test_trinary_data$label)
accuracy_15 = 1-misClassError
print(paste('Accuracy =', accuracy_15))
```

```
## [1] "Accuracy = 0.873804971319312"
```

K=20

```r
classifier_knn_20 <- knn(train = train_trinary_data,test = test_trinary_data,cl = train_trinary_data$la


cm <- table(test_trinary_data$label, classifier_knn_20)
cm
```

```
##    classifier_knn_20
##       0   1   2
##   0 113  18   1
##   1  11 216  13
##   2  10  11 130
```

```r
misClassError <- mean(classifier_knn_20 != test_trinary_data$label)
accuracy_20 = 1-misClassError
print(paste('Accuracy =', accuracy_20))
```

```
## [1] "Accuracy = 0.877629063097514"
```

K=25

```r
classifier_knn_25 <- knn(train = train_trinary_data,test = test_trinary_data,cl = train_trinary_data$la

cm <- table(test_trinary_data$label, classifier_knn_25)
cm
```

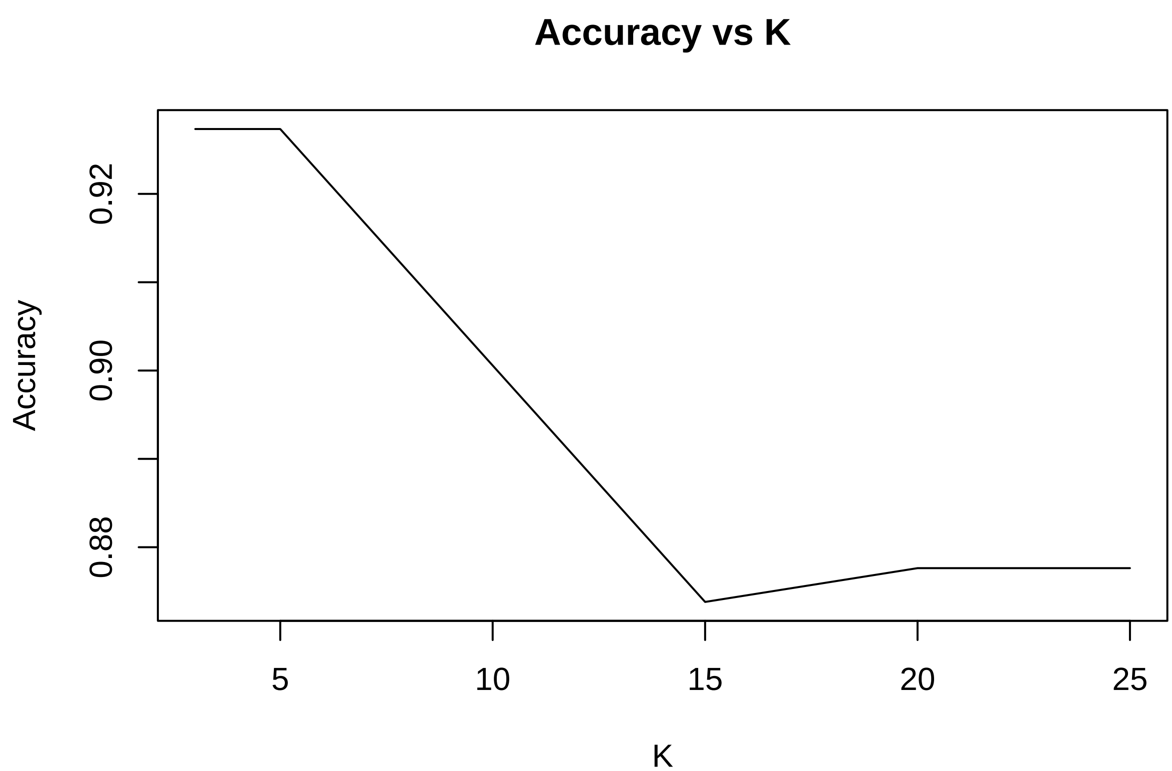
```
##    classifier_knn_25
##       0   1   2
##   0 114  17   1
##   1  11 214  15
##   2  10  10 131
```

```r
misClassError <- mean(classifier_knn_25 != test_trinary_data$label)
accuracy_25 = 1-misClassError
print(paste('Accuracy =', accuracy_25))
```

```
## [1] "Accuracy = 0.877629063097514"
```

Plot for accuray for different K value.

```r
k <- c(3,5,10,15,20,25)
accuracy <- c(accuracy_3,accuracy_5,accuracy_10, accuracy_15, accuracy_20, accuracy_25)
accuracy_by_k <- data.frame(k,accuracy)
plot(accuracy_by_k,type="l",ylab="Accuracy",
 xlab="K",main="Accuracy vs K")
```



As per the plot we can say that K= 3 is giving the optimal accuracy for different k values.

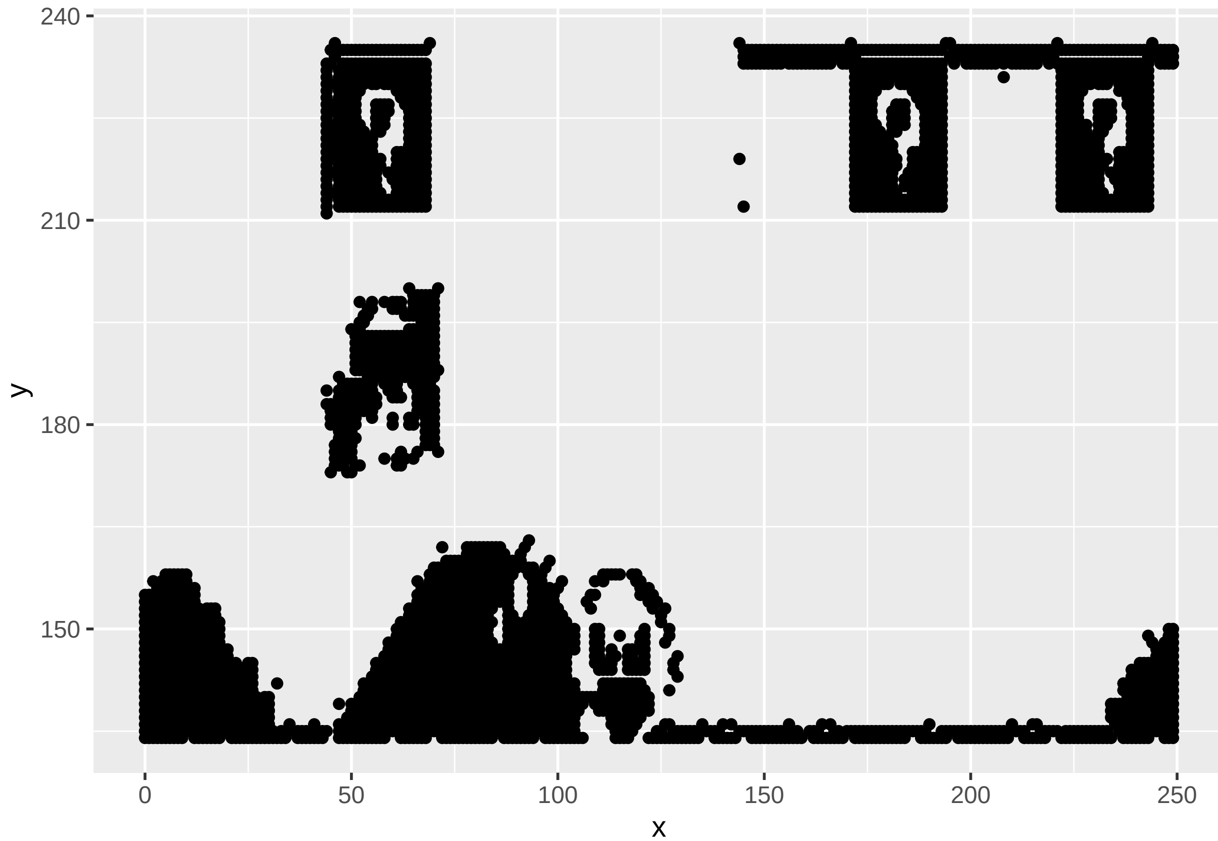
CLUSTERING DATA.

```r
setwd("C:\\Users\\atanu\\Documents\\BellevueUniversity_MSDS\\DSC520\\Repository\\dsc520_")
clustering_data <- read.csv("data\\clustering-data.csv")
head(clustering_data)
```

```
##      x   y
## 1  46 236
## 2  69 236
## 3 144 236
## 4 171 236
## 5 194 236
## 6 195 236
```

Lets see the scatter plot of the data.

```
ggplot(clustering_data, aes(x = x, y = y)) + geom_point()
```

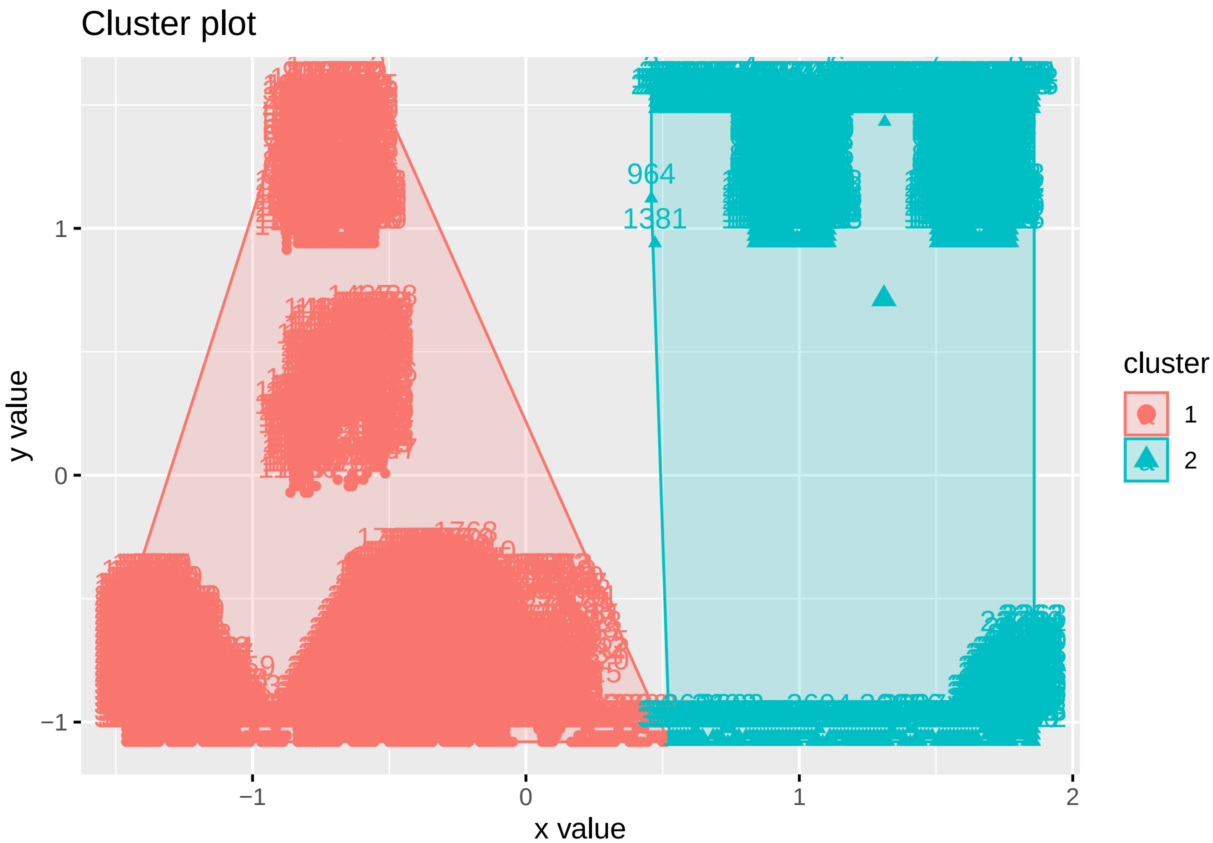


Fitting K Means for K= 2 to K=12

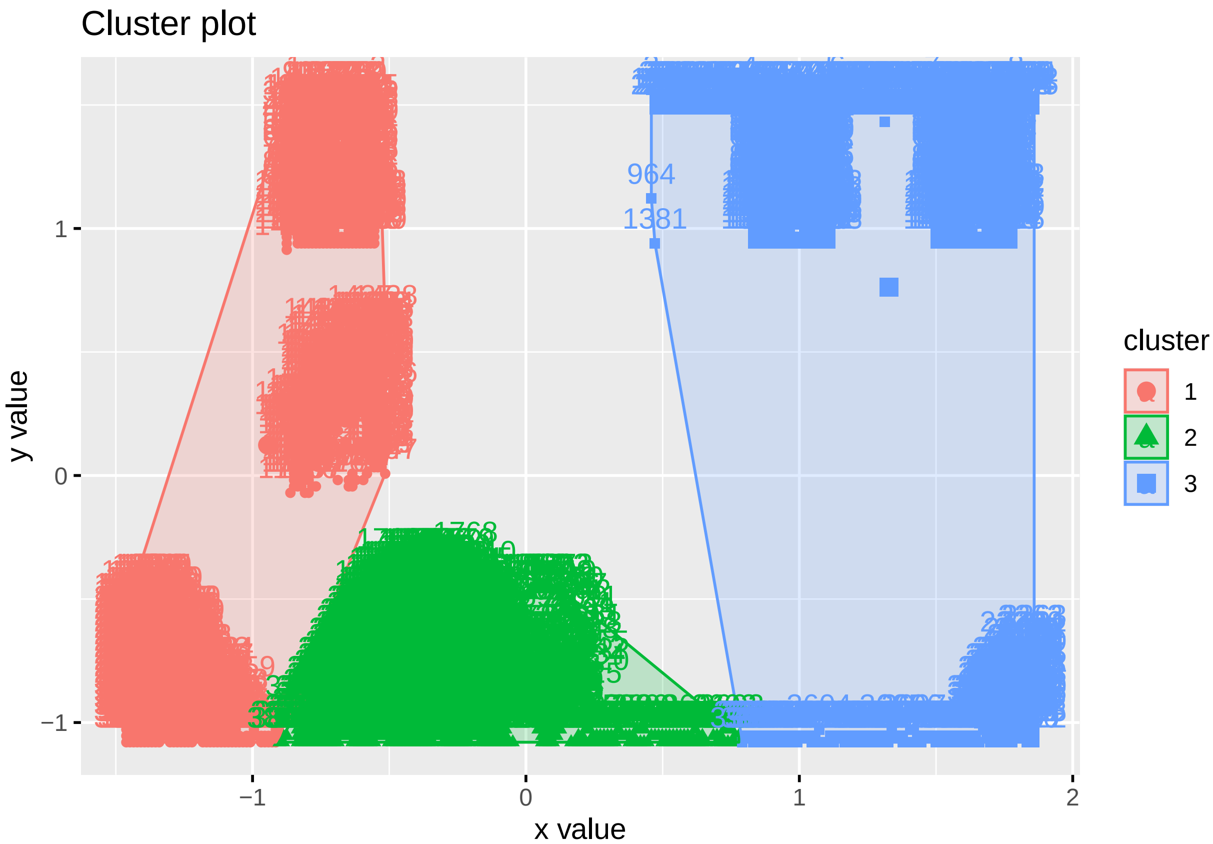
```
library(cluster)    # clustering algorithms
library(factoextra) # clustering algorithms & visualization
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

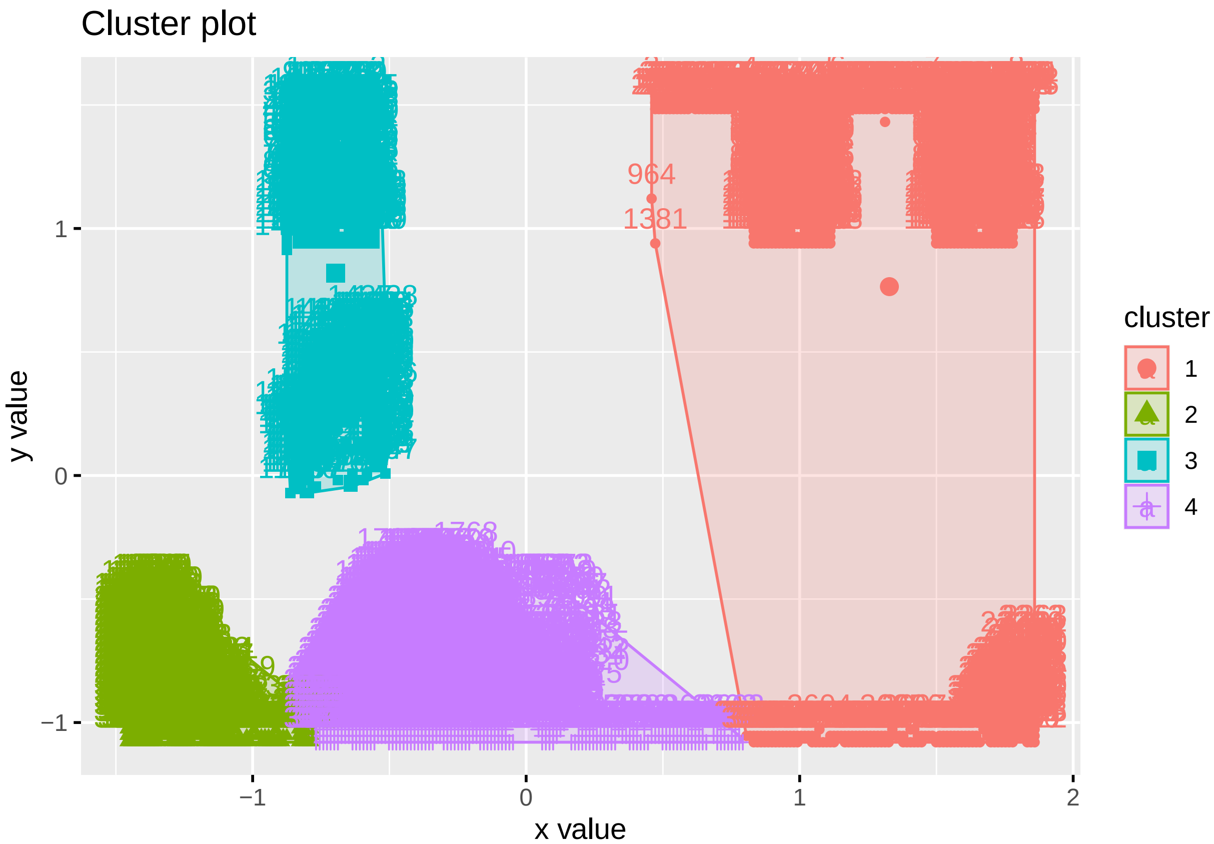
```
k2 <- kmeans(clustering_data, centers = 2, nstart = 25)
fviz_cluster(k2, data = clustering_data)
```



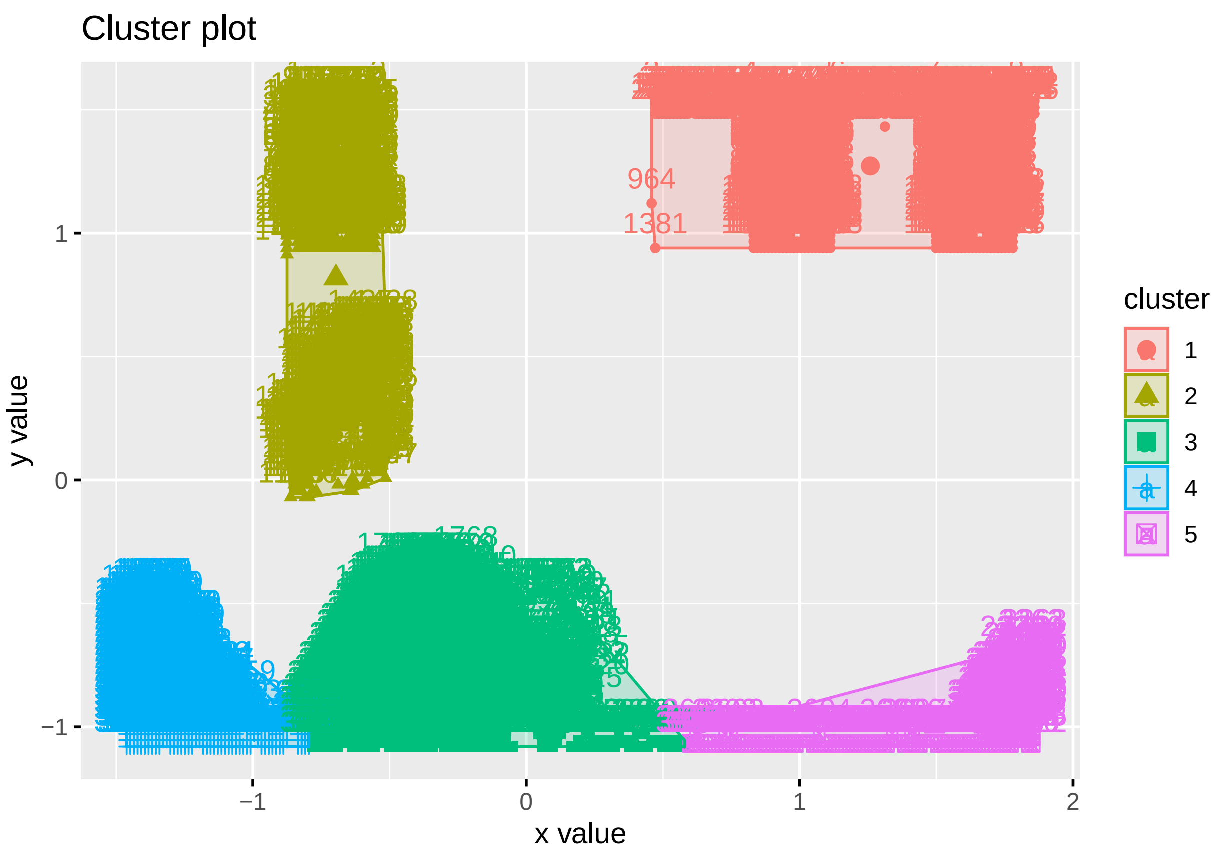

```
k3 <- kmeans(clustering_data, centers = 3, nstart = 25)
fviz_cluster(k3, data = clustering_data)
```



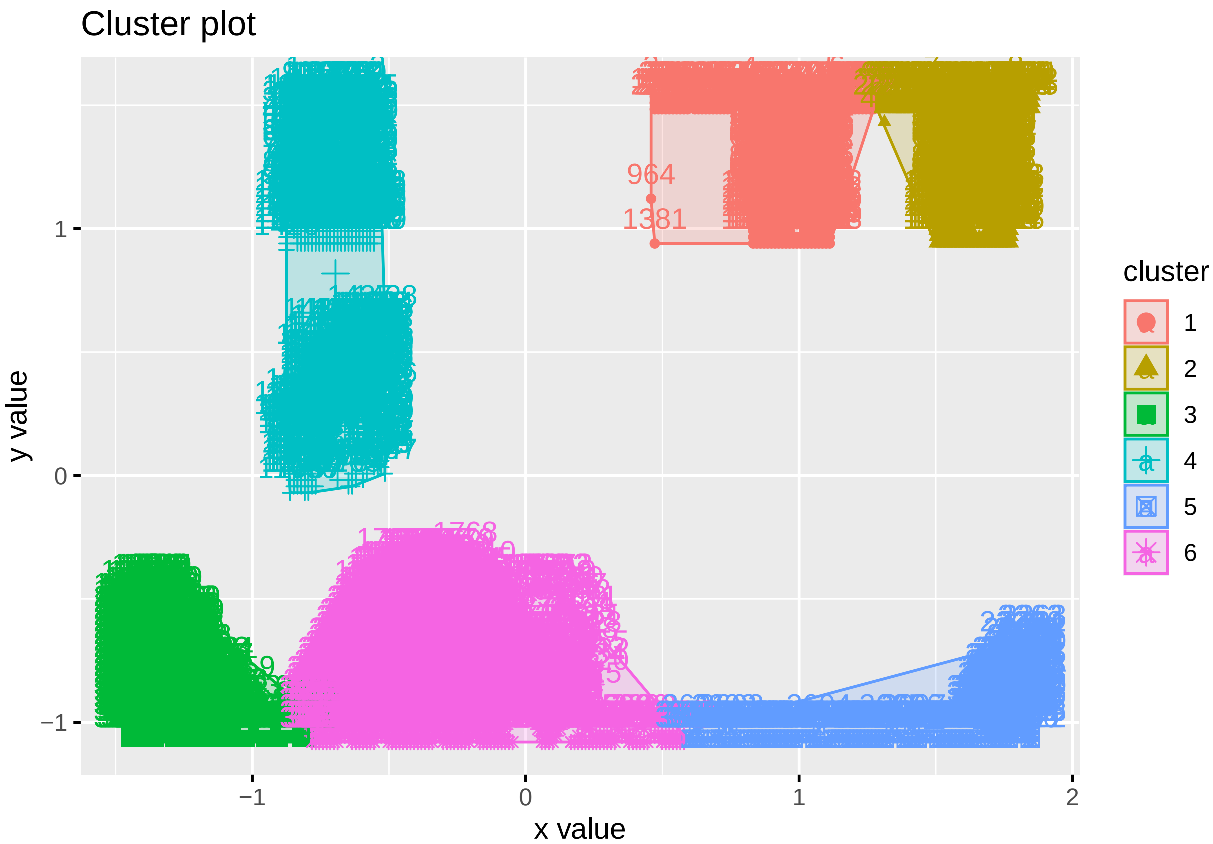

```
k4 <- kmeans(clustering_data, centers = 4, nstart = 25)
fviz_cluster(k4, data = clustering_data)
```



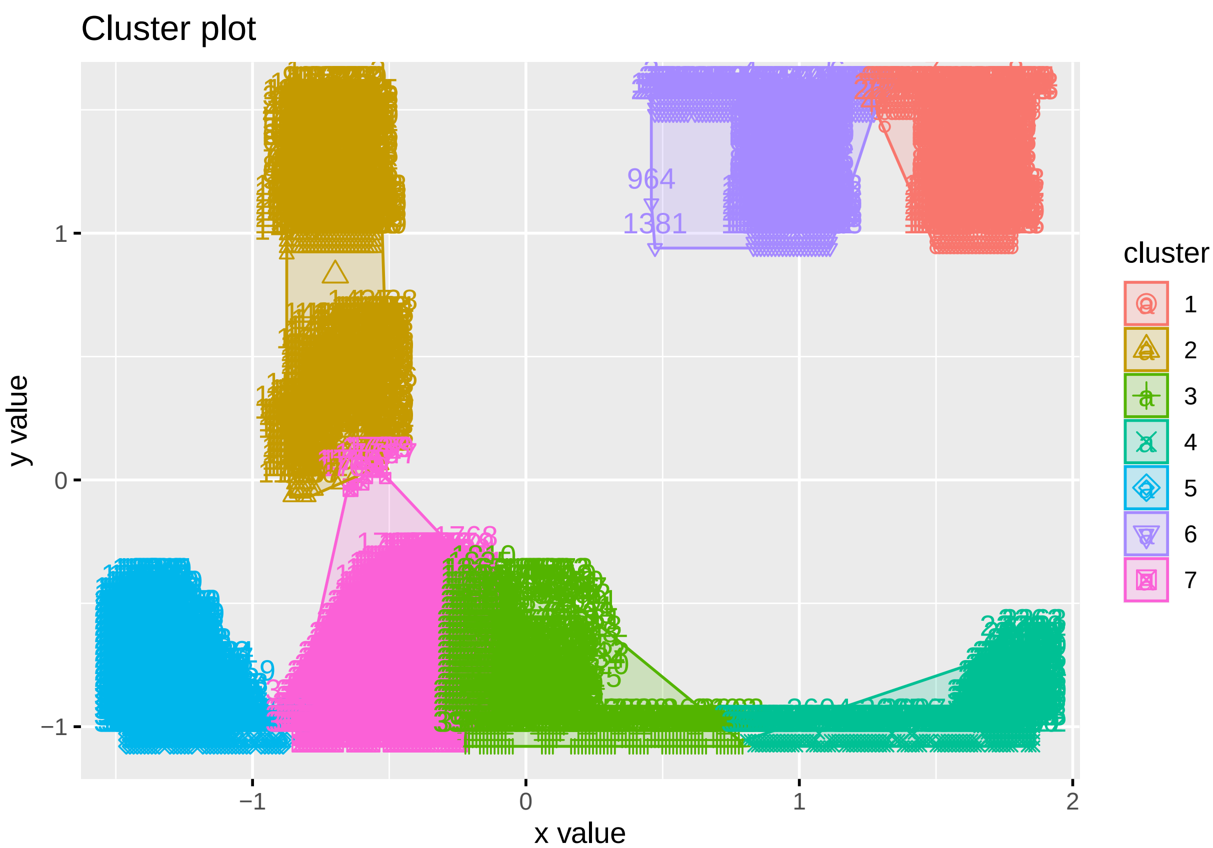

```
k5 <- kmeans(clustering_data, centers = 5, nstart = 25)
fviz_cluster(k5, data = clustering_data)
```



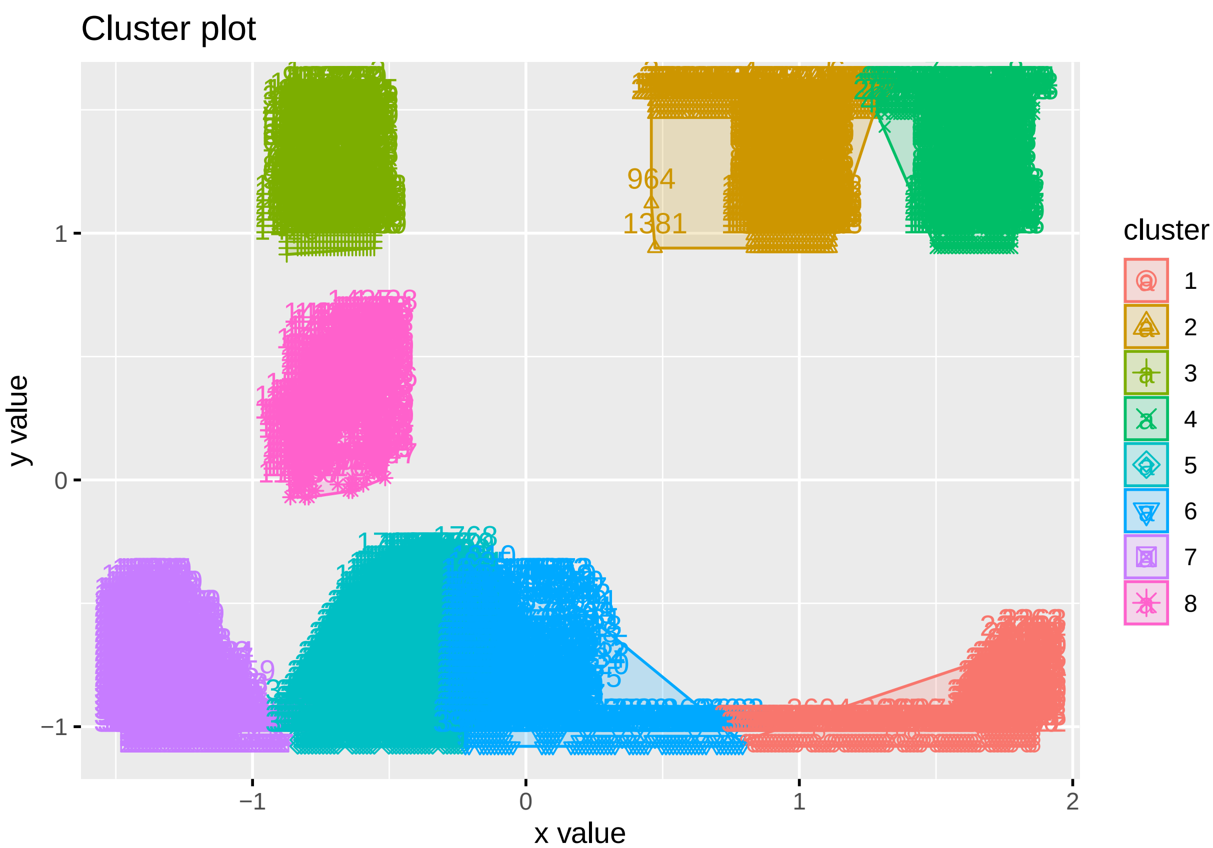

```
k6 <- kmeans(clustering_data, centers = 6, nstart = 25)
fviz_cluster(k6, data = clustering_data)
```



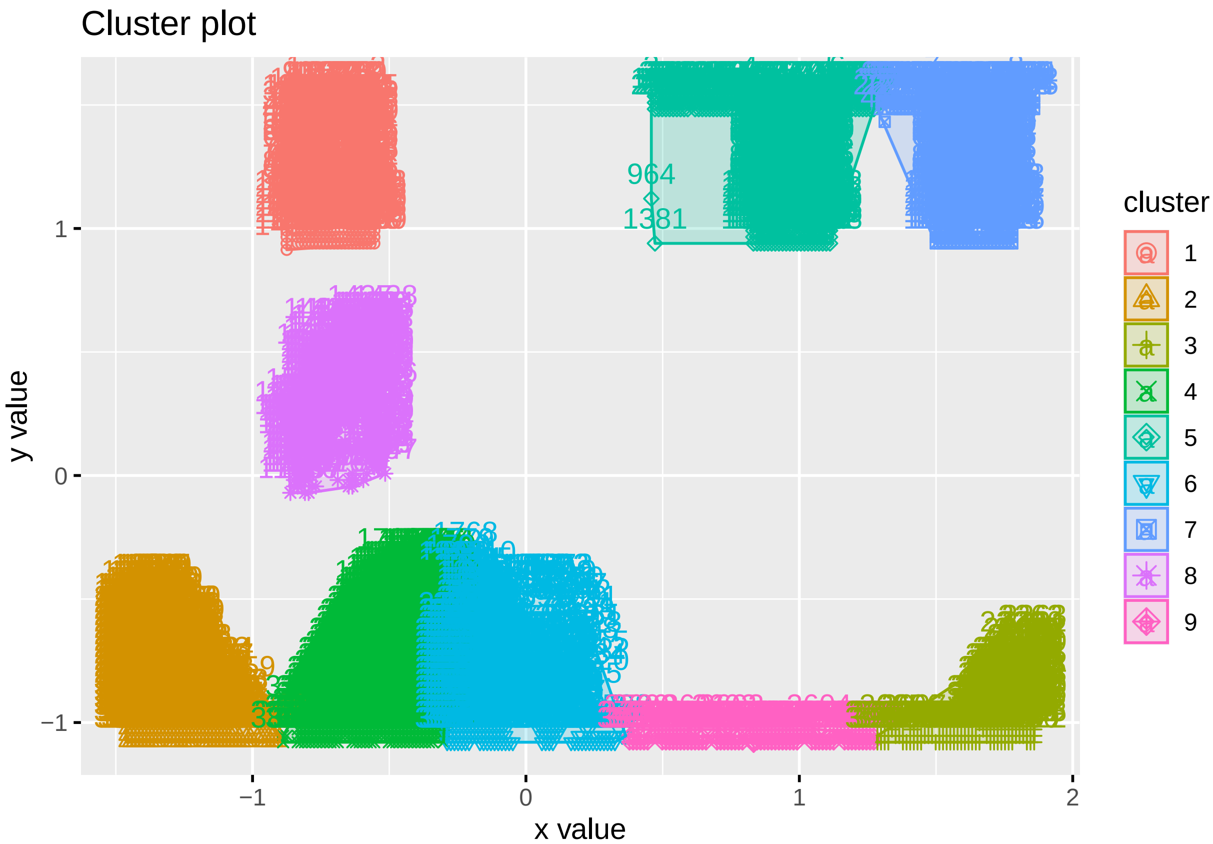

```r
k7 <- kmeans(clustering_data, centers = 7, nstart = 25)
fviz_cluster(k7, data = clustering_data)
```



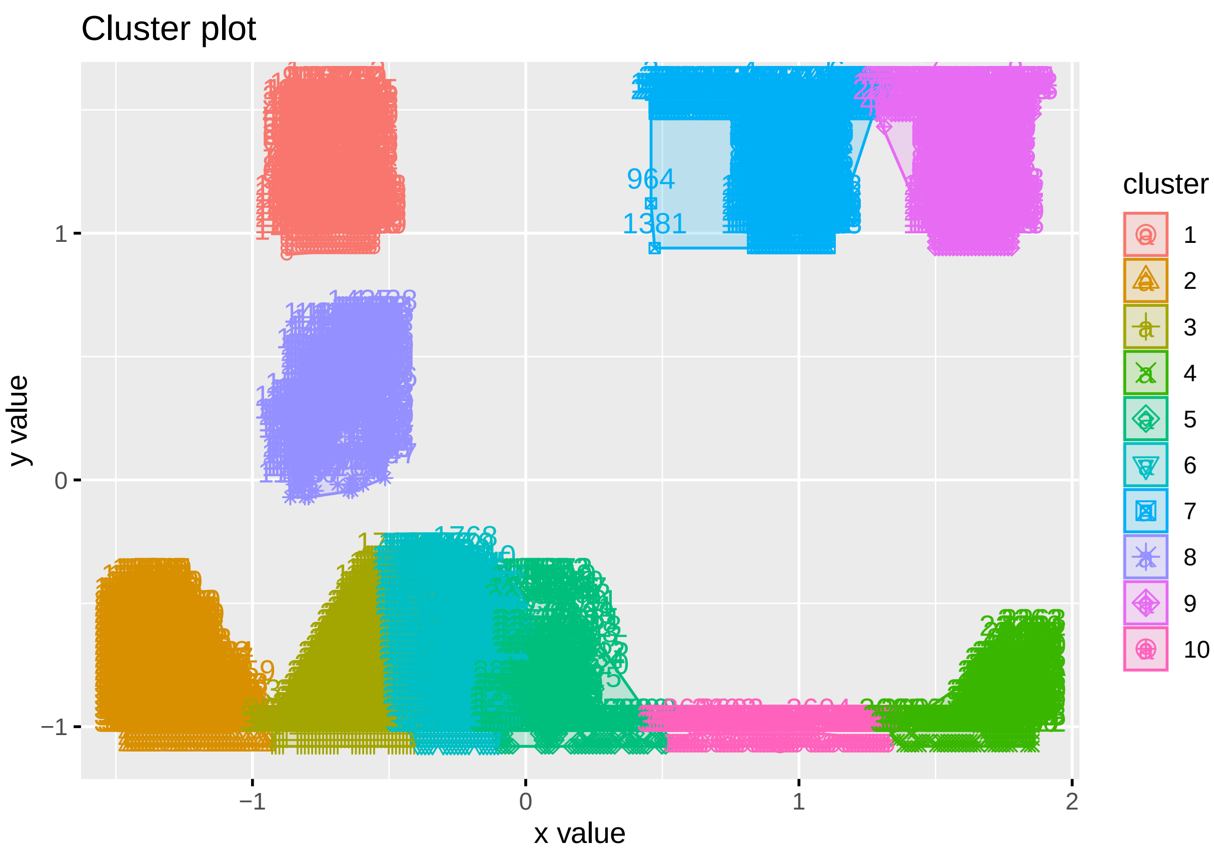

```
k8 <- kmeans(clustering_data, centers = 8, nstart = 25)
fviz_cluster(k8, data = clustering_data)
```



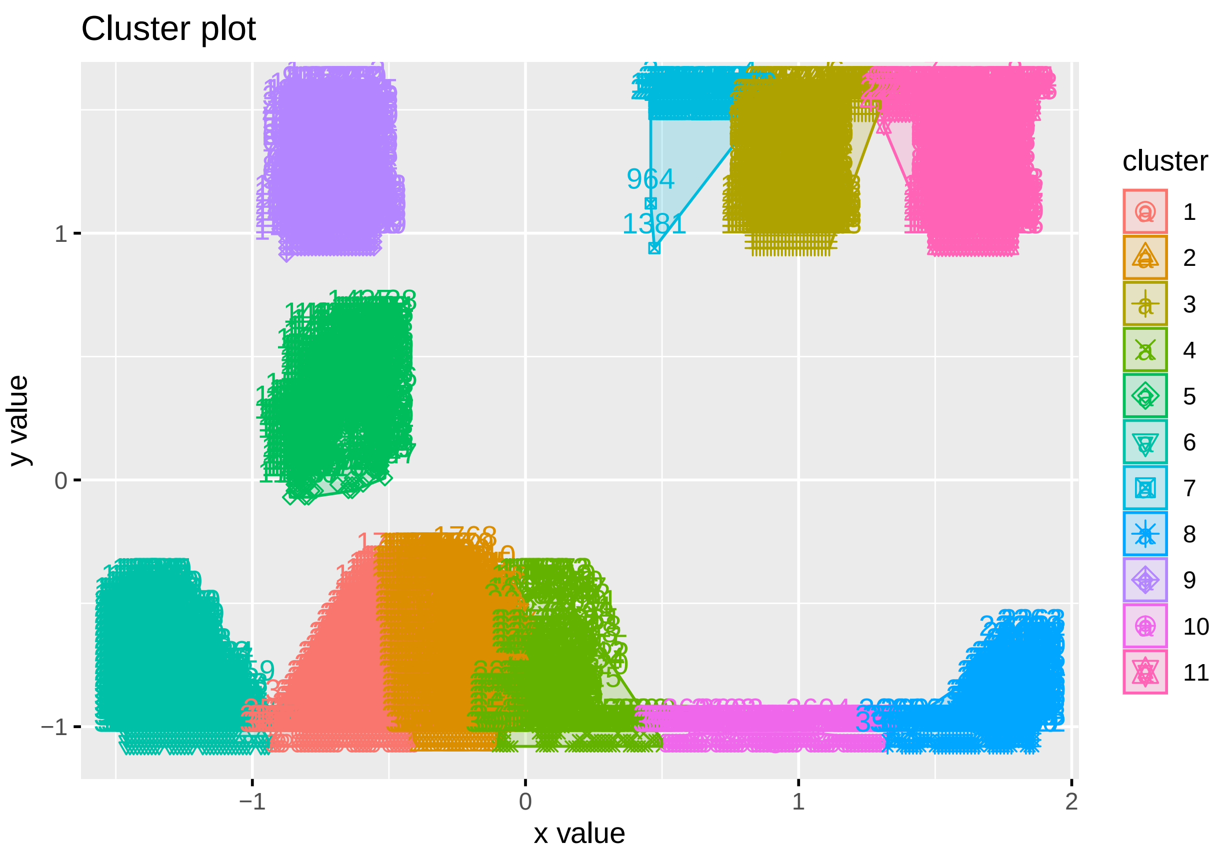

```
k9 <- kmeans(clustering_data, centers = 9, nstart = 25)
fviz_cluster(k9, data = clustering_data)
```



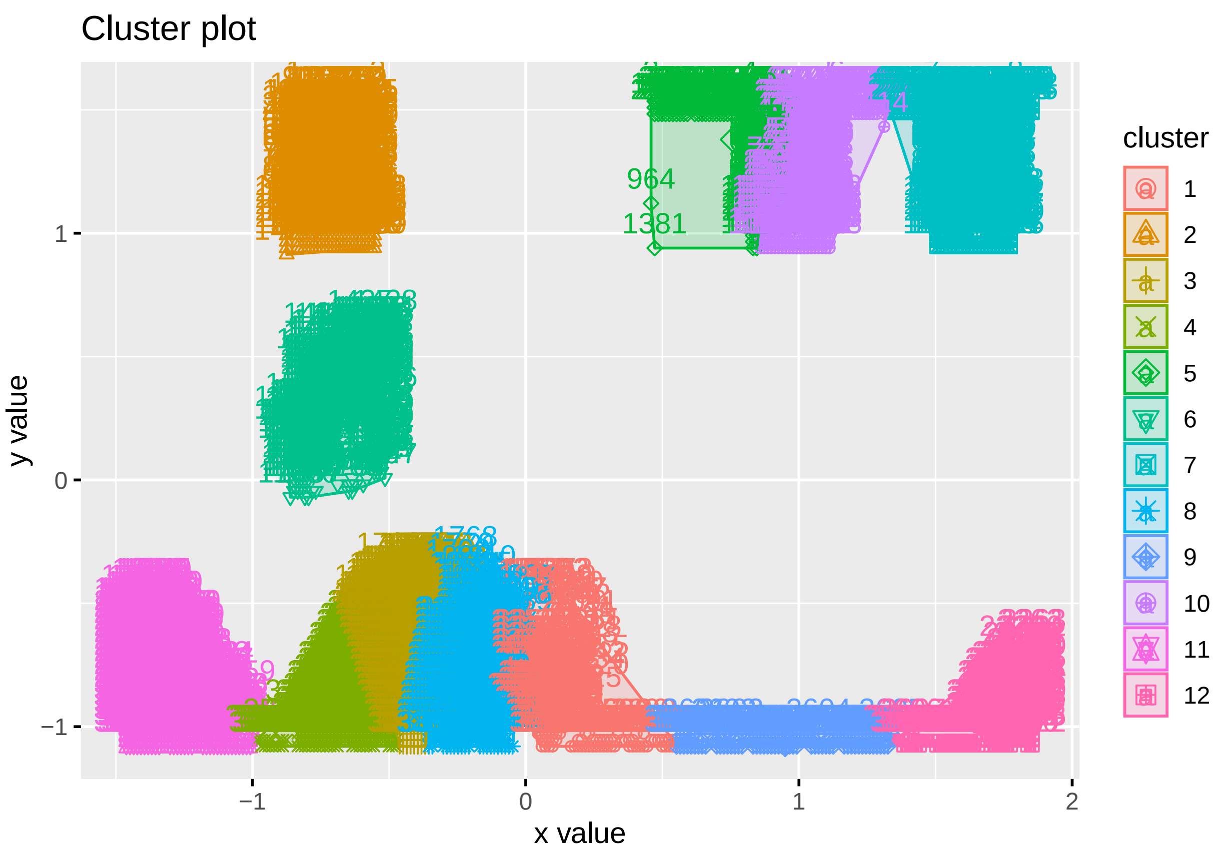

```r
k10 <- kmeans(clustering_data, centers = 10, nstart = 25)
fviz_cluster(k10, data = clustering_data)
```




```
k11 <- kmeans(clustering_data, centers = 11, nstart = 25)
fviz_cluster(k11, data = clustering_data)
```

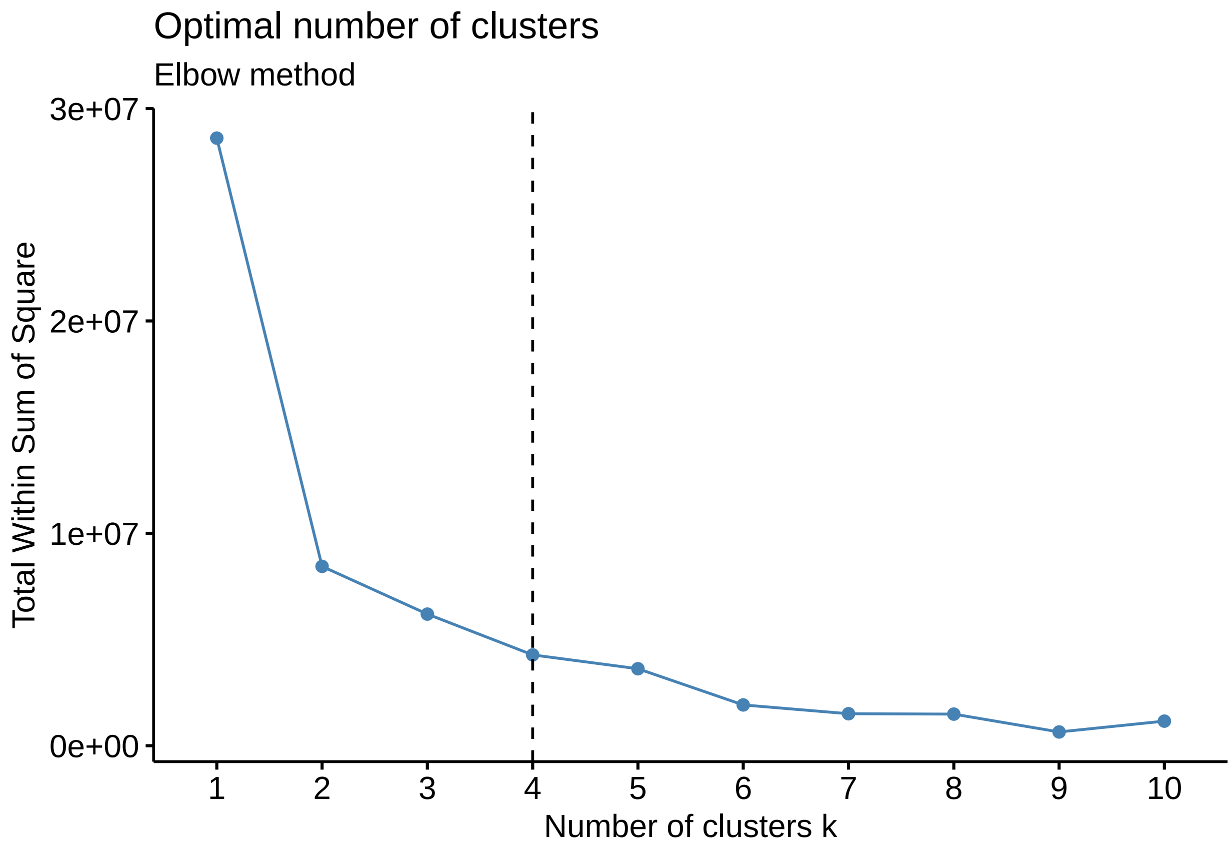



```r
k12 <- kmeans(clustering_data, centers = 12, nstart = 25)
fviz_cluster(k12, data = clustering_data)
```



Lets plot the values of wss (with sum of squares) along with the k and find the elbow to get the optimal K value

```
fviz_nbclust(clustering_data, kmeans, method = "wss") +
  geom_vline(xintercept = 4, linetype = 2) + # add line for better visualisation
  labs(subtitle = "Elbow method") # add subtitle
```



The elbow value in 4 in this case.